

Numerical Study of Impingement Location of Liquid Jet Poured from a Tilting Ladle with Lip Spout



R. CASTILLA, P.J. GAMEZ-MONTERO, G. RAUSH, M. KHAMASHTA,
and E. CODINA

A new approach for simulating liquid poured from a tilting lip spout is presented, using neither a dynamic mesh nor the moving solid solution method. In this case only the tilting ladle is moving, so we propose to rotate the gravitational acceleration at an angular velocity prescribed by a geometrical and dynamical calculation to keep the poured flow rate constant. This angular velocity is applied to modify the orientation of the gravity vector in computational fluid dynamics (CFD) simulations using the OpenFOAM[®] toolbox. Also, fictitious forces are considered. The modified solver is used to calculate the impingement location for six spout geometries and compare the jet dispersion there. This method could offer an inexpensive tool to calculate optimal spout geometries to reduce sprue size in the metal casting industry.

DOI: 10.1007/s11663-017-0920-1

© The Author(s) 2017. This article is published with open access at Springerlink.com

I. INTRODUCTION

THE cast-iron foundry industry produces cast parts for automotive, agriculture, transportation, energy, aerospace, manufacturing industry, *etc.*, with 90 pct of manufactured products containing cast-iron parts. All foundry processes generate a certain level of rejected parts, which are irrecoverably lost, being closely related to the type of casting and processes used and the equipment available. As quality demands from end-users of castings increase, it is essential that cast-iron technology move forward, together with green manufacturing as a first step towards sustainability.

Extensive work has been carried out by the foundry community to minimize casting defects such as porosity, slag, and clogging, but few literature reports consider casting metal using more versatile melt control technology.

The flow control system in a tundish has a significant impact on the quality and level of rejection when pouring casts. If the metal does not flow in a consistent stream, casting defects may result from oxidation, air entrapment, and erosion of the pouring mold, among other causes.

In the cast-iron industry, it is usual for molten metal to be transferred from a ladle with a lip-axis pour design.^[1] This method is fast and reliable, but inaccurate unless a sophisticated control system is used.^[2–4] In recent years, our group has been working on technologies that aim to transform the conventional

(batch-by-batch) foundry process into a flexible (mold-by-mold) process. This requires that the ladle become a furnace as well, that the tundish be eliminated from the process, and that the molten metal be poured directly into the mold. Moreover, it is convenient for the pouring flow rate to be constant to obtain uniform refilling of the mold. Such a flexible mold-by-mold process is characterized by, firstly, combination of the melting, treatment, and pouring processes into a single cast-iron production cell. Also, this requires integration of an artificial-intelligence-based control system to monitor local structures, phases, and mechanical properties to guarantee high-quality casting in the foundry. Finally, a robot cell will be in charge of the metal finishing process. This also reduces the melt temperature and transport while improving validation of the cast pieces.

In recent decades, extensive effort has been invested in development of CFD simulation methods for application in the casting industry, especially for multiphase flows.^[5–7] In the particular case of simulations of pour tilt casting, Prakash *et al.*^[8] used the smoothed particle hydrodynamics method to simulate the oxidation process during furnace emptying, and Kuriyama *et al.*^[9] used the software Flow 3D to optimize the pouring velocity in aluminum tilt casting. Pauty *et al.*^[10] carried out two-dimensional (2D) numerical simulations of the liquid transport and thermal convective velocity in a tilting furnace, finding good agreement with water experiments, while admitting that the extension of the method to three-dimensional (3D) flow simulations of molten metal would require a dramatic increase of computational resources and time. Fortunately, advances in computer technologies in recent decades have made this kind of complex 3D fluid flow simulation more affordable; For instance, Davila *et al.*^[11] computed the 3D flow during drainage of a ladle to understand vortex mechanisms in this process.

R. CASTILLA, P. J. GAMEZ-MONTERO, G. RAUSH, and E. CODINA are with the LABSON, Department of Fluid Mechanics, Universitat Politècnica de Catalunya, 08222, Terrassa, Spain. Contact e-mail: castilla@mf.upc.edu M. KHAMASHTA is with the LABSON, Department of Mechanical Engineering, Universitat Politècnica de Catalunya, 08222, Terrassa, Spain.

Manuscript submitted July 26, 2016.

We present herein a study of the impingement location of liquid poured from a tilting ladle having a lip spout with different curvatures. It has been reported that a curved lip provides better jet control and reduces air entrainment.^[12] The purpose of this work is to present a numerical method with low computer power requirements to calculate the liquid trajectory in such a pour tilt casting process. The results could be used to reduce jet dispersion to minimize liquid spilling and reduce sprue size.

Section II develops the theoretical background to calculate the angular velocity required to achieve constant flow rate. The results are the angular velocity and tilting angle *vs* time. In the next stage, these data are used in CFD simulations. The numerical approach is described in Section III. The simulations were performed using the OpenFOAM[®] toolbox, which is based on the finite-volume method (FVM). As this is a dynamic process with moving boundaries, the habitual strategy is to use a dynamic mesh. Another option is the FAVOR[®] technique.^[13,14] These two methods are very flexible and allow independent movement of several objects in the domain. However, we propose an alternative method where neither the mesh nor object move, but the gravitational acceleration. This method is simpler and faster than the dynamic mesh or FAVOR[®] approach, but has the disadvantage that only one moving object is allowed (in this case, the tilting ladle) and the postprocessing is more complicated. The results are presented in Section IV; finally, in Section V, these results are discussed and conclusions drawn.

II. THEORETICAL BACKGROUND

The theoretical calculation of the flow rate poured from the furnace is based on conservation of mass of liquid. The flow rate of liquid is equal to the time derivative of the mass inside the ladle, thus

$$\frac{\partial}{\partial t} \int_{VC} \rho dV = -\rho q, \quad [1]$$

where q is the flow rate through the spout. Since the density ρ is constant, it can be eliminated from Eq. [1]. Assuming that the free surface of the liquid always remains horizontal, the volume of liquid can be divided into two regions: the volume above a horizontal plane passing through the lower point of the liquid exit surface, V_t , and the volume below this plane, V_b . The height of V_t is h , so that

$$V_t = h A_{FS}, \quad [2]$$

where A_{FS} is the area of the free surface. Figure 1 depicts a scheme of the pouring liquid with arbitrary tilting angle θ .

Thus, Eq. [1] yields

$$\frac{dV_t}{dt} = -q - \frac{dV_b}{dt}, \quad [3]$$

which can be expressed in terms of the angle θ instead of time, thus

$$-\omega \frac{dV_t}{d\theta} = -q + \omega \frac{dV_b}{d\theta}, \quad [4]$$

where $\omega = -d\theta/dt$ is the angular tilting velocity of the ladle. Using Eq. [2] in [4] yields

$$A_{FS} \frac{dh}{d\theta} + h \frac{dA_{FS}}{d\theta} = \frac{q}{\omega} - \frac{dV_b}{d\theta}. \quad [5]$$

The lower volume, V_b , is known to be a cylindrical wedge and can be calculated based on geometric arguments if the spout lip is neglected (*i.e.*, assuming that the ladle is a perfect cylinder). If the free surface does not cut the cylinder (ladle) bottom, the volume can be easily calculated as^[15]

$$V_b = \frac{1}{2} \pi R^2 (H + H_s) = \pi R^2 H \left(1 - \frac{R}{H \tan \theta} \right). \quad [6]$$

If, otherwise, the plane cuts the cylinder bottom, the calculation is more complex. In the particular case in which the plane cuts the bottom circle across its diameter, known then as a cylindrical hoof, the volume

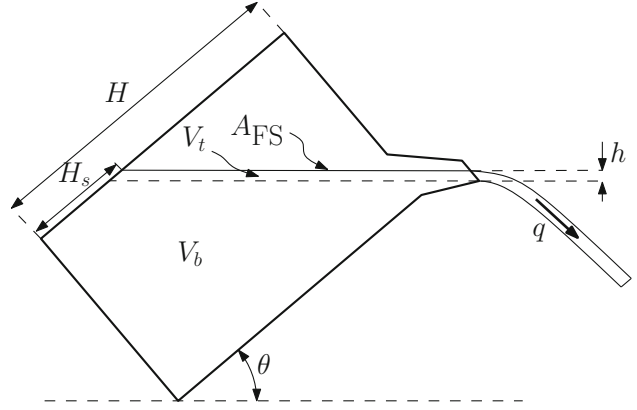


Fig. 1—Scheme for pouring liquid for arbitrary tilting angle θ (not to scale).

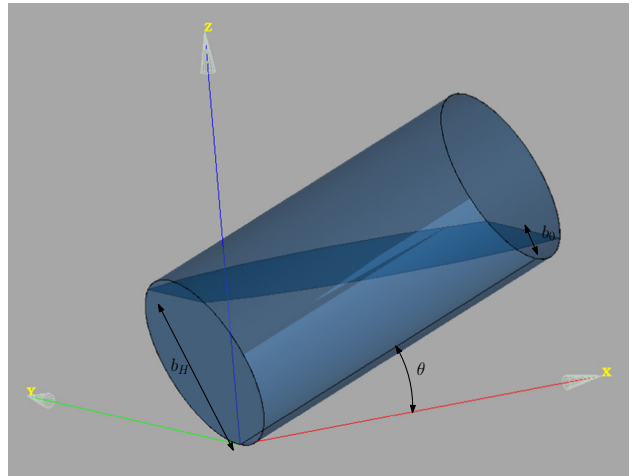


Fig. 2—Scheme of partially full inclined cylinder.

is one-sixth the volume of the square prism in which the cylinder is inscribed, as shown by Archimedes^[16]

$$V_b = \frac{2}{3} R^2 H. \quad [7]$$

For the general case, the volume has to be calculated by integration, giving

$$V_b = \frac{R^3}{\tan \theta} \left[\pi(\eta_H - \eta_0) - \eta_H \arccos \eta_H + \frac{1}{3} \sqrt{1 - \eta_H^2} (2 + \eta_H^2) + \eta_0 \arccos \eta_0 - \frac{1}{3} \sqrt{1 - \eta_0^2} (2 + \eta_0^2) \right], \quad [8]$$

where

$$\eta_0 = \frac{b_0}{R} - 1, \quad [9]$$

$$\eta_H = \eta_0 + \frac{H \tan \theta}{R}, \quad [10]$$

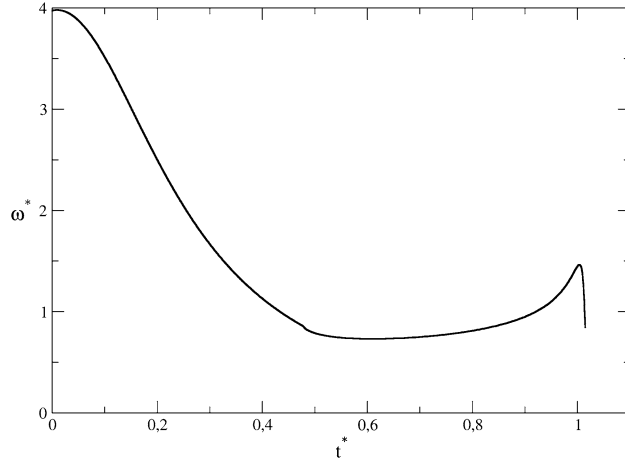


Fig. 3—Theoretical estimation of dimensionless angular velocity vs dimensionless time for constant flow rate.

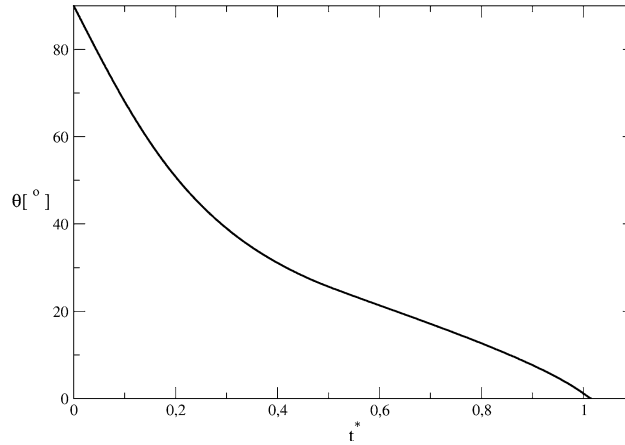


Fig. 4—Theoretical estimation of rotation angle vs dimensionless time for constant flow rate.

where b_0 is the height of the liquid on the opposite side of the cylinder (see Figure 2). For our case of pouring liquid, $b_0 = 0$ and Eq. [8] reduces to

$$V_b = \pi H R^2 - \frac{R^3}{\tan \theta} \left[\pi + \eta \arccos \eta - \frac{1}{3} \sqrt{1 - \eta^2} (2 + \eta^2) \right], \quad [11]$$

where $\eta = H \tan \theta R - 1$. Alternatively, one can use the following expression from Reference 15:

$$V_b = \frac{1}{3} H R^2 \left[\frac{3 \sin \phi - 3 \phi \cos \phi - \sin^3 \phi}{1 - \cos \phi} \right], \quad [12]$$

where

$$\phi = \frac{\pi}{2} + \arctan(\eta \sqrt{1 - \eta^2}).$$

Both equalities, *i.e.*, Eqs. [11] and [12], reduce to the Archimedes theorem, Eq. [7], in the case of the cylindrical hoof (*i.e.*, $\eta = 0$ in Eq. [11]).

The area of the free surface is trivial when the plane does not cut the bottom of the cylinder, being an ellipse with area

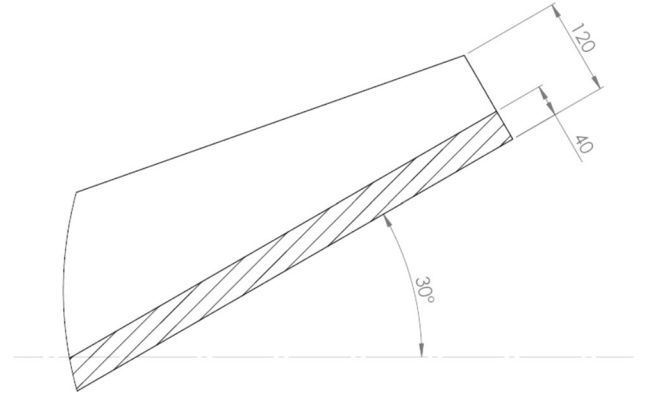


Fig. 5—Original spout geometry (lengths in mm).

$$A_{FS} = \frac{\pi R^2}{\sin \theta}, \quad [13]$$

but is more complicated when the plane cuts the bottom. In this case, it is a segment of an ellipse, and, according to Hugues *et al.*,^[17] can be calculated as

$$A_{FS} = \frac{R^2}{\sin \theta} (\psi - \sin \psi), \quad [14]$$

where

$$\psi = 2 \arccos(-\eta).$$

The flow rate in Eq. [5] can be estimated using the expression for the flow rate over a circular weir^[18]

$$q = 0.3926 C_d \sqrt{2gh^{3/2}} D \beta^{1/2} \left(\sqrt{1 - 0.2200\beta} + \sqrt{1 - 0.7730\beta} \right), \quad [15]$$

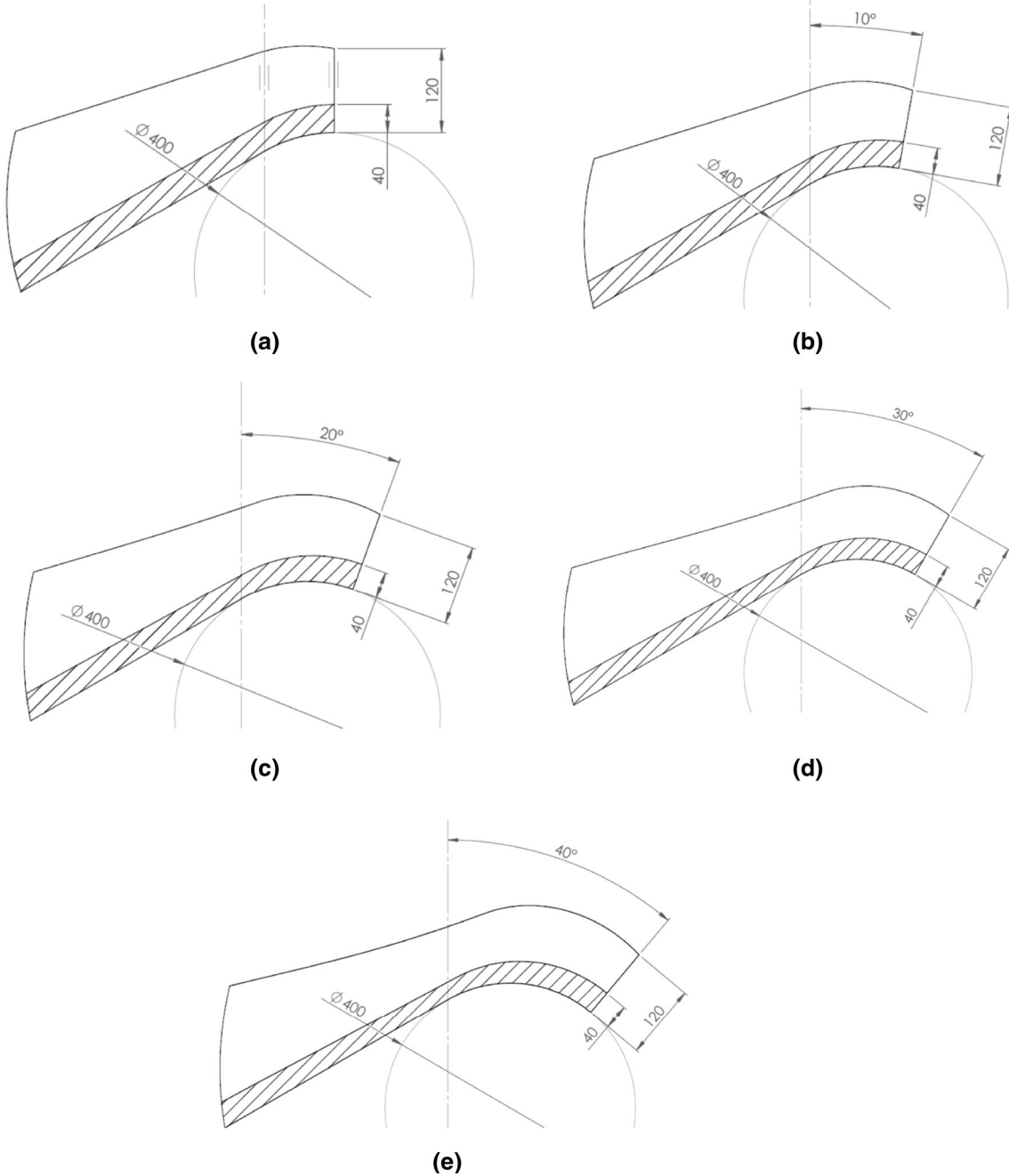


Fig. 6—Geometries of simulated spouts (lengths in mm) with spout angle of (a) 0 deg, (b) 10 deg, (c) 20 deg, (d) 30 deg, and (e) 40 deg.

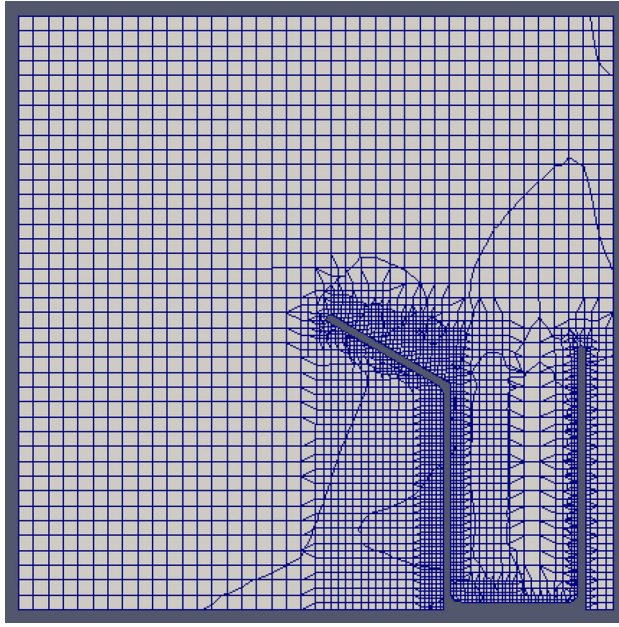


Fig. 7—Slice of mesh for original geometry.

where $\beta = h/D$ and C_d is the discharge coefficient, which is also a function of β . Nevertheless, a simplified expression^[19] is usually applied:

$$q = KD^{n_1} h^{n_2}, \quad [16]$$

where $n_1 = 0.693$, $n_2 = 1.807$, and $K = 1.598$ when S.I. units are used.

Given the flow rate poured, which is kept constant, Eq. [16] leads to the height of the free surface as

$$h = \left(\frac{q}{KD^{n_1}} \right)^{\frac{1}{n_2}}. \quad [17]$$

Thus, the angular velocity as a function of angle can be calculated from Eq. [5], provided that the flow rate q and the free surface height h are constant,

$$\omega(\theta) = \frac{q}{h \frac{dA_{FS}}{d\theta} + \frac{dV_b}{d\theta}}, \quad [18]$$

where the derivative of the free surface area is calculated from Eqs. [13] and [14], and the derivative of the bottom volume is obtained from Eqs. [6] and [11].

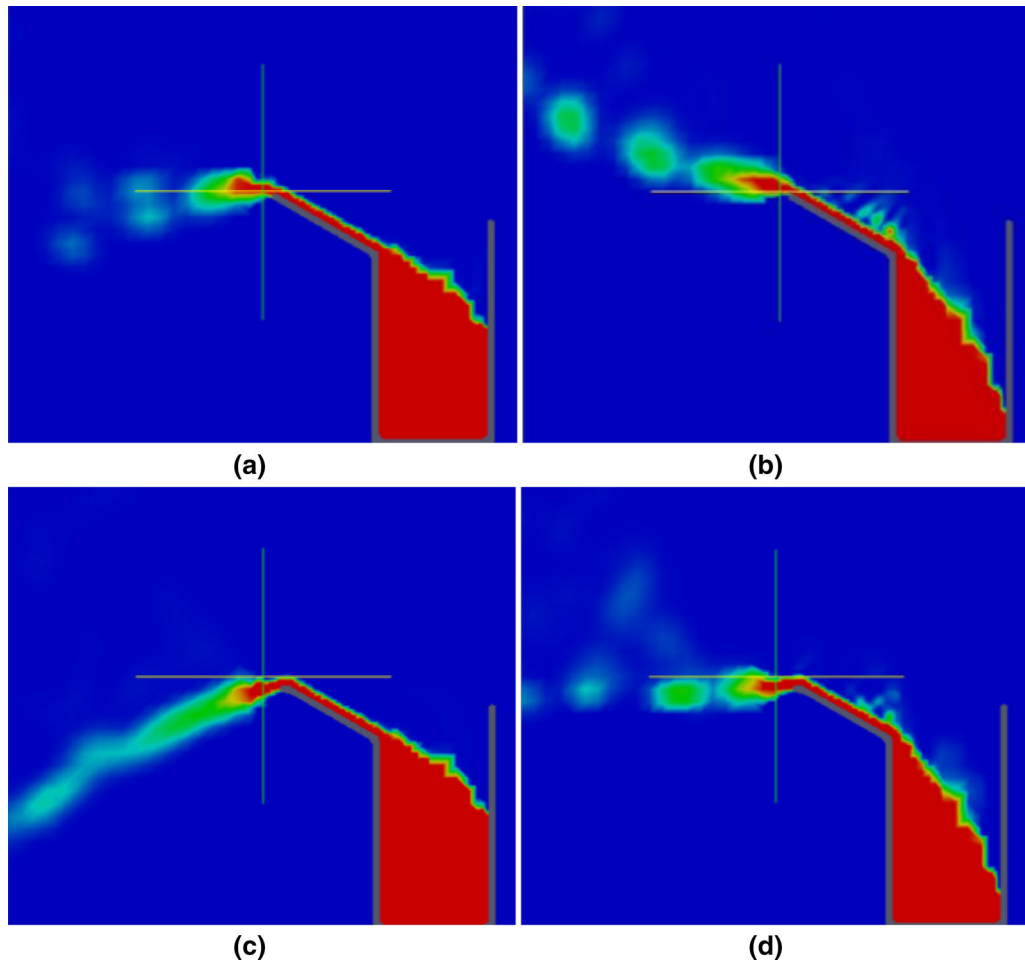


Fig. 8—Simulated pouring for $t^* = 0.3$ and $t^* = 0.6$ for original case (a, b) and simulation 1 with spout angle of 0 deg (c, d).

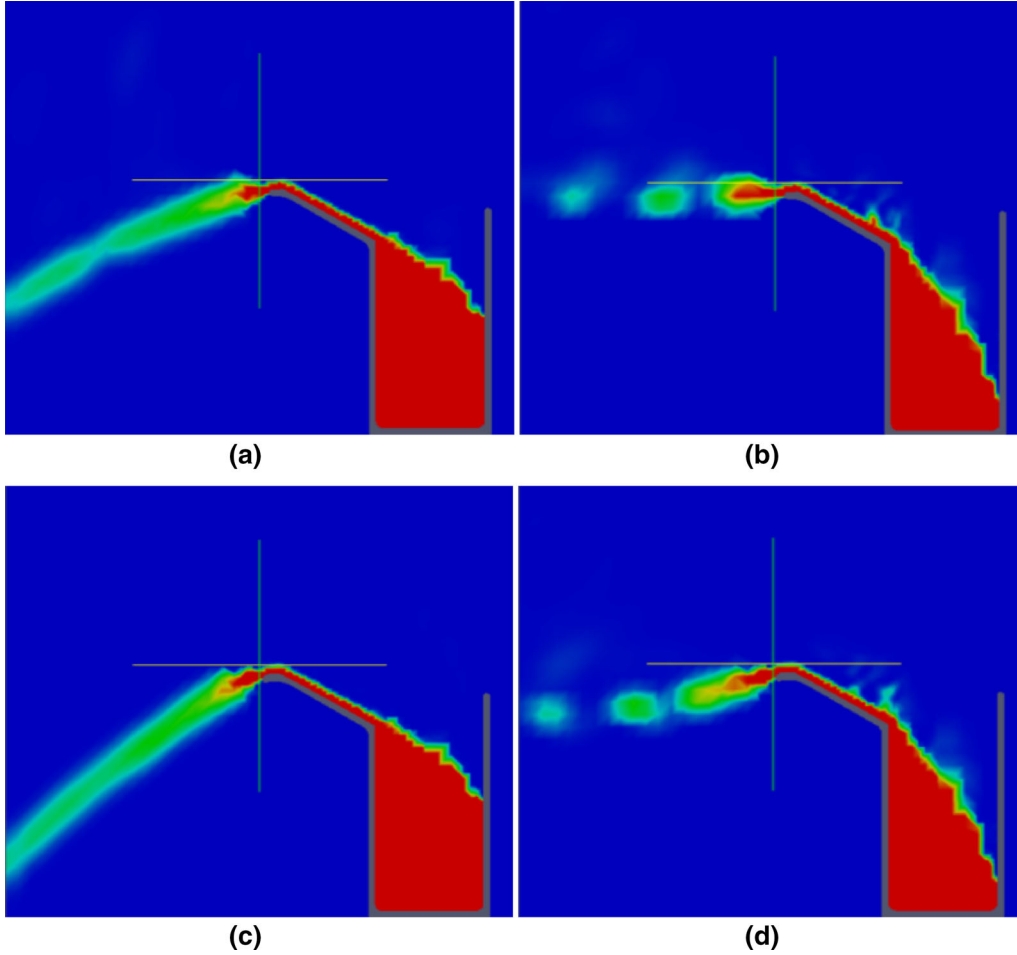


Fig. 9—Simulated pouring for $t^* = 0.3$ and $t^* = 0.6$ for simulation 2 with spout angle of 10 deg (a, b) and simulation 3 with spout angle of 20 deg (c, d).

This angular velocity can be expressed in terms of time, instead of angle, using

$$t(\theta) = \int_{\frac{\pi}{2}}^{\theta} \frac{d\alpha}{\omega(\alpha)}. \quad [19]$$

Combining Eqs. [18] and [19], one obtains the angular velocity ω vs time. The dimensionless angular velocity and time can be defined using a characteristic time $\tau = \pi R^3 \lambda q$, with a form factor $\lambda = \frac{H}{R}$, which is the time needed to empty the cylinder at flow rate q ,

$$\begin{aligned} \omega^* &= \omega \tau \\ t^* &= t \tau. \end{aligned}$$

Figure 3 shows the dimensionless angular velocity, calculated from Eq. [18], against dimensionless time. Figure 4 shows the rotation angle vs dimensionless time.

III. NUMERICAL APPROACH

CFD simulations were carried out using the OpenFOAM[®] toolbox.^[20–22] This open-source tool provides

libraries, solvers, applications, and utilities to solve partial differential equations (PDEs) using the FVM. There are several methods for solving multiphase flows, but the most popular is the volume-of-fluid (VOF) approach,^[23] in which a local liquid fraction γ is defined in each cell domain. This liquid fraction defines a cell as completely filled with liquid ($\gamma = 1$) or gas ($\gamma = 0$), or partially occupied by liquid and gas ($0 < \gamma < 1$). The liquid fraction is transported with the flow as

$$\frac{\partial \gamma}{\partial t} + \nabla \cdot (\gamma \vec{u}) + \nabla \cdot (\gamma(1 - \gamma) \vec{u}_r) = 0, \quad [20]$$

where the last term on the left-hand side represents interface compression and avoids dispersion of interfaces, keeping a narrow zone of cells with $\gamma \approx 0.5$. For this purpose, the velocity \vec{u}_r is artificially defined normal to the interface and pointing towards it.^[24]

Along with the transport equation for the liquid fraction, the conservation of mass and momentum are solved, *i.e.*,

$$\nabla \cdot \vec{u} = 0, \quad [21]$$

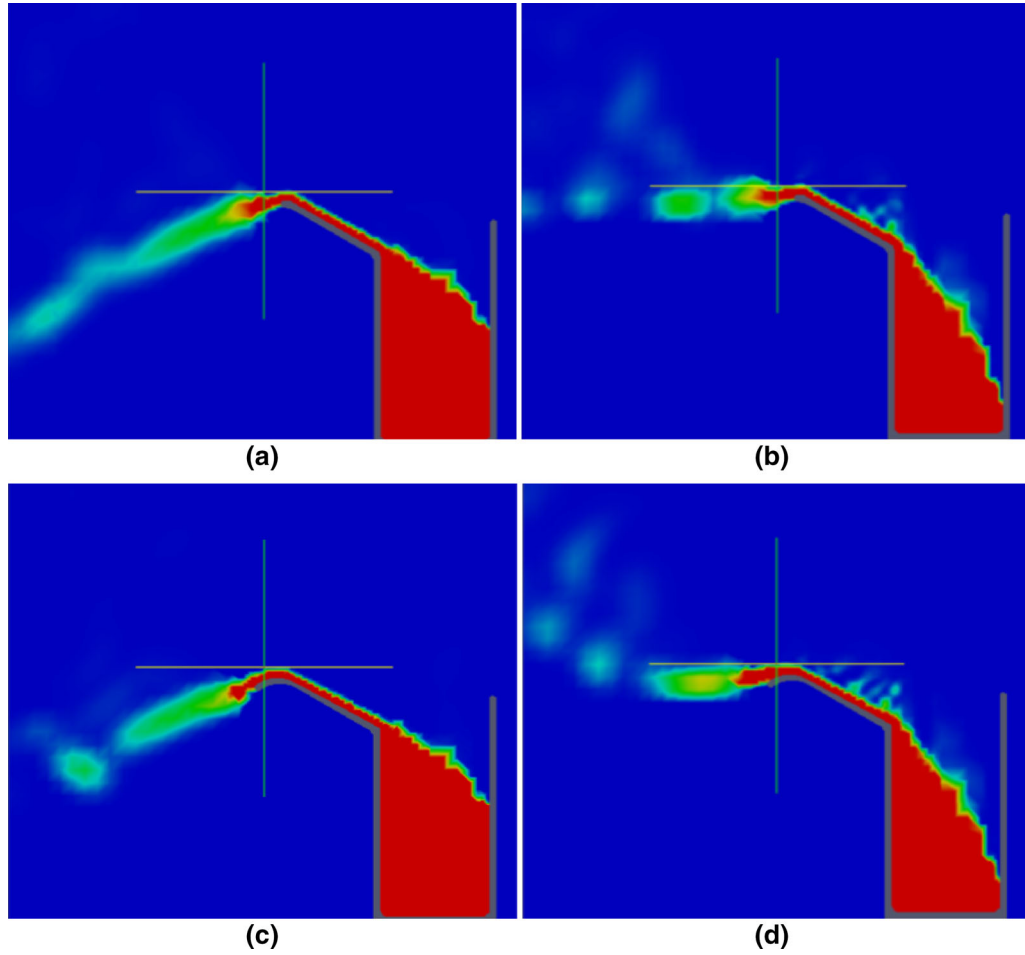


Fig. 10—Simulated pouring for $t^* = 0.3$ and $t^* = 0.6$ for simulation 4 with spout angle of 30 deg (a, b) and simulation 5 with spout angle of 40 deg (c, d).

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u} \vec{u}) = -\nabla p + \nabla \cdot (\mu (\nabla \vec{u} + \nabla \vec{u}^T)) + \rho \vec{g} + \int_{S_i} \sigma \kappa \vec{n} \delta(\vec{r} - \vec{r}_i) dS, \quad [22]$$

where \vec{g} is the gravitational field, σ is the surface tension coefficient, κ is the curvature of the interface, and \vec{n} is the unitary vector normal to the interface surface S_i . The density ρ and viscosity μ in each cell are computed as

$$\rho = \gamma \rho_l + (1 - \gamma) \rho_g, \quad [23]$$

$$\mu = \gamma \mu_l + (1 - \gamma) \mu_g, \quad [24]$$

where the subscripts “l” and “g” indicate the liquid and gas phase, respectively.

From the point of view of computational resources and simplicity, it is better to keep the ladle steady and rotate the gravitational field. This requires modification of the solver so that it calculates the tilting angle according to the angular velocity history, as shown in

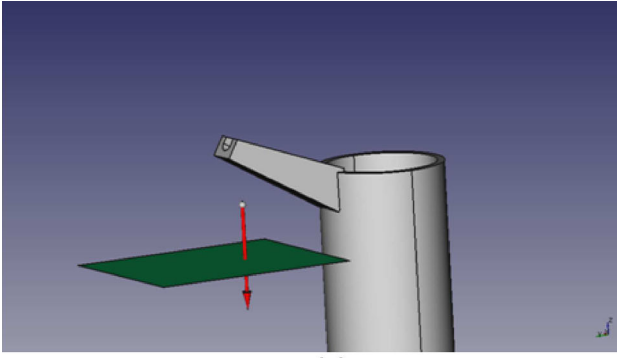
Figure 3, and modifies the direction of the gravitational vector in the momentum Eq. [22].

This also implies that the computational domain is a noninertial frame of reference, hence fictitious forces due to rotation must also be included in the momentum equation. This fictitious acceleration, which has to be included in the right-hand side of the momentum equation, is

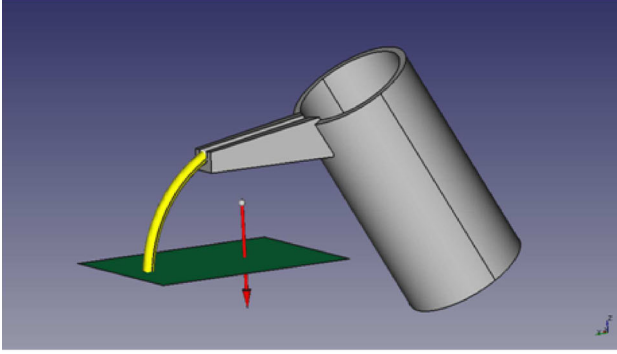
$$\vec{a}_f = -\vec{r} \times \frac{d\vec{\omega}}{dt} - 2\vec{\omega} \times \vec{u} - \vec{\omega} \times \vec{\omega} \times \vec{r}. \quad [25]$$

Although in the typical case in the present study all terms in this equation are much smaller than gravity, they were included in the simulations, except the first one (rotational acceleration), which is typically three orders of magnitude smaller than the other two (Coriolis and centrifugal forces).

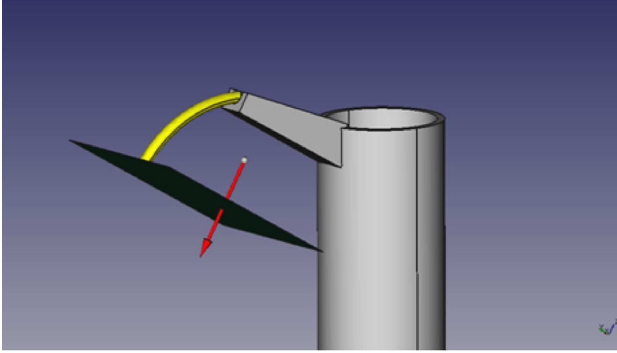
With respect to the last term in the momentum Eq. [22], the problem is that, in the VOF method, the interface is not explicitly tracked, so the integral in this term cannot be computed directly. Brackbill^[25] overcame this problem by using a continuum surface force model which considers the surface tension force as a volumetric force that acts only on the interface as



(a)



(b)



(c)

Fig. 11—Scheme of rotation of impingement plane as a function of pouring angle: (a) vertical tank, (b) tank tilted by 45 deg, and (c) vertical tank with impingement plane and gravity rotated by 45 deg.

$$\int_{S_i} \sigma \kappa \vec{n} \delta(\vec{r} - \vec{r}_i) dS \approx \sigma \kappa \nabla \gamma, \quad [26]$$

where the curvature κ is computed as

$$\kappa = \nabla \cdot \left(\frac{\nabla \gamma}{\|\nabla \gamma\|} \right), \quad [27]$$

so that it vanishes away from the interface.

The time derivative term is computed using an Euler implicit scheme, which is first order and unconditionally stable. The spatial discretization schemes are second order with Gaussian integration. Since the objective of this study is to simulate the jet trajectory, which is supposed to be not much affected by turbulence

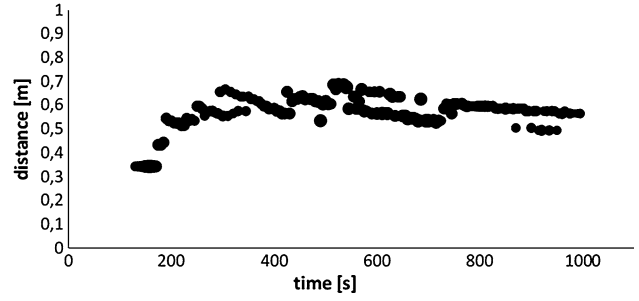


Fig. 12—Impingement position as function of time for original geometry.

structures, no turbulence model was explicitly included in the simulations.

Five ladle geometries, besides the original one, were numerically analyzed. The ladle inner cylinder had height of 0.97 m and base radius of 0.244 m. The original spout geometry is shown in Figure 5. Several lip inclination angles, with respect to the cylinder vertical, were studied for spout curvature of 200 mm. The angles simulated were 0, 10, 20, 30, and 40 deg. The geometries are shown in Figure 6. The ladle rotates around an axis located at $y = 0.9$ m, $z = 1.2$ m, with the center of coordinates at the center of the ladle bottom.

All meshes were created using snappyHexMesh. The meshes have around 95,000 cells, most of them (about 70 pct) being orthogonal hexahedra (Figure 7). The simulations were run on an AMD Opteron 6100 Ubuntu cluster with 64 cores and 64 GB RAM. Each simulation (1000 seconds) required around 25 h with domain decomposition of 16 cores.

IV. RESULTS

As mentioned above, this study focuses on the trajectory of the poured liquid and, especially, on the impingement location of the liquid on the tundish baseline. Figures 8 through 10 show the trajectory of the liquid for the original case and the five geometric modifications for $t^* = 0.3$ and $t^* = 0.6$. There is a notable difference in the trajectory as a function of spout angle. To quantify this difference for all times against the impingement baseline, further postprocessing calculations must be performed.

The impingement baseline was located 1 m above the tank base, that is, 200 mm below the center of rotation. We define a line with this point and normal to the gravity vector, *i.e.*, dependent on time.

The *sample* utility, shipped with the OpenFOAM package, allows computation of the liquid fraction γ along this line. Using a python script, this line was calculated at each time, using the same angle data as the solver, and input to the *sample* utility. The distribution of γ along this line was thereby obtained for each time. This procedure is shown schematically in Figure 11. From this distribution of γ along the rotated x axis, the position of the maximum value of γ was sought and recorded for each time. The result of this computation for the original

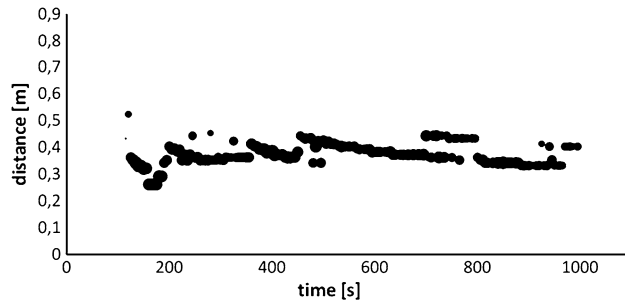


Fig. 13—Impingement position as function of time for simulation 1 (spout angle 0 deg).

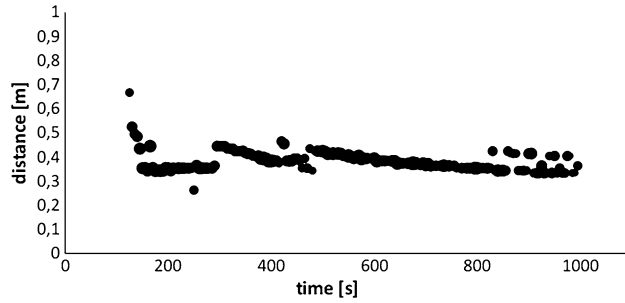


Fig. 14—Impingement position as function of time for simulation 2 (spout angle 10 deg).

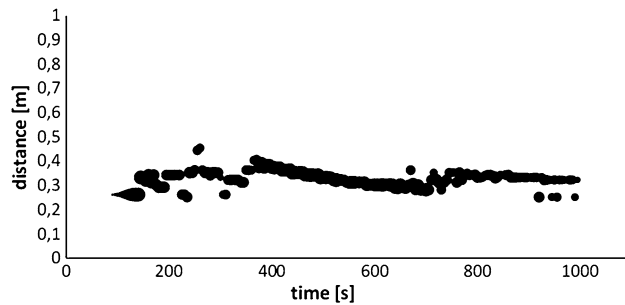


Fig. 15—Impingement position as function of time for simulation 3 (spout angle 20 deg).

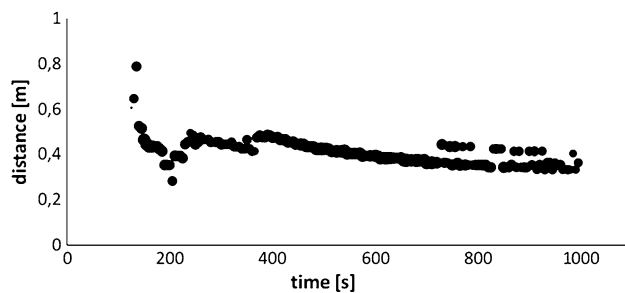


Fig. 16—Impingement position as function of time for simulation 4 (spout angle 30 deg).

(straight) spout is plotted in Figure 12. The size of the points is proportional to the maximum value of γ .

The impingement position of the poured jet remained more or less stable at $x \approx 0.6$ m after the first 300 seconds. However, at the beginning of the process, the

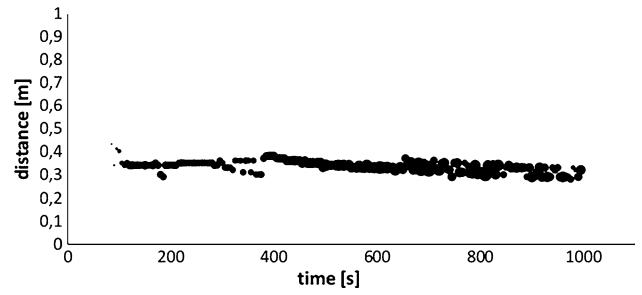


Fig. 17—Impingement position as function of time for simulation 5 (spout angle 40 deg).

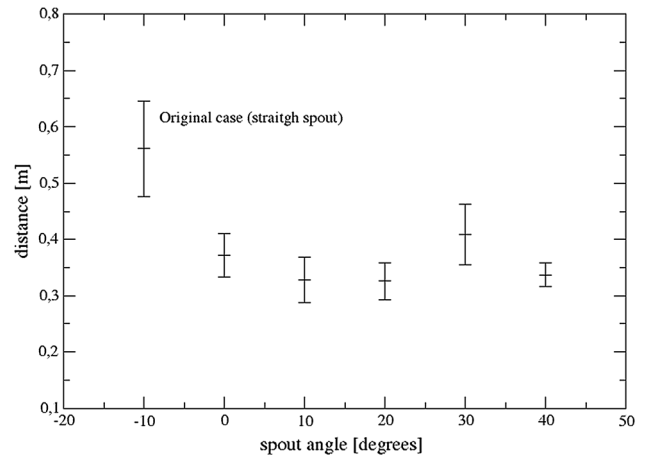


Fig. 18—Impingement location and extension of impact zone for original case and five simulated cases.

dispersion is quite large, with the first impact occurring at around $x \approx 0.3$ m. The full extent of the pouring jet on the impingement line is around 170 mm (we consider the extent of the jet to be twice the weighted standard deviation).

When the spout was curved up to 0 deg, the behavior was better, as seen in Figure 13. The dispersion was narrower and closer to the center of rotation, with average position at around $x \approx 350$ mm and extension of 190 mm. Figures 14 through 17 show the impingement location of the liquid for simulations 2 to 5.

These results are gathered and plotted in Figure 18 as the mean position of the impingement location and its standard deviation vs spout angle. The average is weighted by the value of γ at each point as well as the standard deviation. The original case (straight spout) is also displayed for comparison.

From these results, it can be concluded that, among the tested cases, the best option is the last one (simulation 5) with spout angle of 40 deg. In this case, the extension of the impingement location is 90 mm.

V. DISCUSSION AND CONCLUSIONS

Control of the impingement location of a jet poured from a tilting ladle is important in the molten metal manufacturing industry, to avoid spilling liquid outside of the

correct location. This work numerically studied its passive control by modifying the angle of the lip spout of the ladle.

Firstly, the angular velocity of the tank required for constant flow rate was calculated, based on geometric considerations and the flow rate over a circular weir. The resulting angular velocity was used as input for a set of CFD simulations to calculate the trajectory of the poured liquid for six cases, viz. the original geometry (straight spout) and five angles (0, 10, 20, 30, and 40 deg) with 200 mm radius of curvature. To simplify the fluid dynamics computations, we proposed to modify the solver to consider variable gravity, rotated at the previously calculated angular velocity. This avoids use of a dynamic mesh, which is computationally expensive. The disadvantage is that the postprocessing is more complicated, since the impingement plane also has to be rotated with gravity.

The results are presented as the γ -weighted average of the impingement location and its standard deviation vs spout angle. These results showed that, among the tested cases, the last one with angle of 40 deg was best, having extension of the impingement location four times smaller than for the original case with a straight spout.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the computer resources at Magerit and the technical support provided by the Supercomputing and Visualisation Center of Madrid (CeSViMa) (FI-2016-1-0011). The authors disclose receipt of the following financial support for the research, authorship, and/or publication of this article: This research program has received funding from the European Union Seventh Framework Programme (FP7/2007–2013) under grant agreement no. 314540.

OPEN ACCESS

This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

REFERENCES

1. E. Neumann and D. Trauzeddel: *Foundry Trade J. (UK)*, 2002, vol. 176, pp. 23–24.
2. Y. Noda and K. Terashima: *IFAC Proc. Vol.*, 2007, vol. 40, pp. 53–58.
3. A. Ito, Y. Noda, K. Terashima in *Control Applications (CCA)*, 2012 *IEEE International Conference on*, IEEE, pp. 246–51.
4. K. Terashima and K. Yano: *Control Eng. Pract.*, 2001, vol. 9, pp. 607–20.
5. H. Appa, D. Deglon, and C. Meyer: *Hydrometallurgy*, 2013, vols. 131–132, pp. 67–75.
6. H. Appa, D. Deglon, and C. Meyer: *Hydrometallurgy*, 2014, vols. 147–148, pp. 234–40.
7. T. Kumaresan and S. Thakre: *Hydrometallurgy*, 2014, vol. 150, pp. 107–22.
8. M. Prakash, P. Cleary, and J. Grandfield: *J. Mater. Process. Technol.*, 2009, vol. 209, pp. 3396–407.
9. K.Y.Y. Kuriyama and S. Nishido: Optimization of Pouring Velocity for Aluminium, in *Fluid Dynamics, Computational Modeling and Applications*, L.H. Juarez, ed., InTech, Rijeka, 2012, pp. 575–88.
10. E. Pauty, B. Laboudigue, and J.A.A. Etay: *Metall. Mater. Trans. B*, 2000, vol. 31, pp. 207–14.
11. O. Davila, L. Garcia-Demedices, and R.D. Morales: *Metall. Mater. Trans. B*, 2006, vol. 37, pp. 71–87.
12. J.D. Walker: *An Innovative New Pouring Design for Steel Castings*, ProQuest, Dubai, 2008.
13. C. Hirt, J. Sicilian in *International Conference on Numerical Ship Hydrodynamics, 4th* (Washington, DC).
14. G. Wei: *Mod. Phys. Lett. B*, 2005, vol. 19, pp. 1719–22.
15. J.W. Harris and H. Stöcker: *Handbook of Mathematics and Computational Science*, Springer, New York, 1998.
16. N. Palladino, in *The Genius of Archimedes – 23 Centuries of Influence on Mathematics, Science and Engineering: Proceedings of an International Conference held at Syracuse, Italy, June 8-10, 2010*, eds. By A. S. Paipetis, M. Ceccarelli, An Archimedean Research Theme: The Calculation of the Volume of Cylindrical Groins (Springer, Dordrecht, 2010), pp. 3–15.
17. G.B. Hughes and M. Chraibi: *Comput. Vis. Sci.*, 2014, vol. 15, pp. 291–301.
18. A.R. Vatankhah: *Flow Meas. Instrum.*, 2010, vol. 21, pp. 118–22.
19. J. de Azevedo Netto: *Manual de hidráulica*, Harla, Mexico City, 1976.
20. H. Weller, G. Tabor, H. Jasak, and C. Fureby: *Comput. Phys.*, 1998, vol. 12, p. 12.
21. H. Jasak, Ph.D. thesis, Imperial College of Science, Technology and Medicine, 1996.
22. C.J. Greenshields: *OpenFOAM-The Open Source CFD Toolbox-Programmer's Guide*, 2nd ed., OpenFOAM Foundation Ltd., London, 2015.
23. C. Hirt and B. Nichols: *J. Comput. Phys.*, 1981, vol. 39, p. 201.
24. H. Rusche, Ph.D. thesis, Imperial College, London, 2002.
25. J. Brackbill, D. Kothe, and C. Zemach: *J. Comput. Phys.*, 1992, vol. 100, p. 335.